Measuring Relative Accuracy of Malware Detectors in the Absence of Ground Truth

John Charlton
Depart. of Computer Science
UT San Antonio

Pang Du
Depart. of Statistics
Virginia Tech

Jin-Hee Cho
Depart. of Computer Science
Virginia Tech

Shouhuai Xu
Depart. of Computer Science
UT San Antonio

***Abstract*—In this paper, we measure the *relative* accuracy of malware detectors in the *absence* of ground truth regarding the quality of malware detectors (i.e., the detection accuracy) or the class of sample files, (i.e., malicious or benign). In particular, we are interested in measuring the *ordinal* scale of malware detectors in the absence of the ground truth of their actual detection quality. To this end, we propose an algorithm to estimate the relative accuracy of the malware detectors. Based on synthetic data with known ground truth, we characterize when the proposed algorithm leads to accurately estimating the relative accuracy of the malware detectors. We show the measured relative accuracy of real-world malware detectors using our proposed algorithm based on a real dataset consisting of 10.7 million files and 62 malware detectors, obtained from VirusTotal.**

***Index Terms*—Malware detection, security metrics, security measurement, ground truth, estimation, statistical estimators.**

I. Introduction

Measuring security metrics is a vital but challenging open research problem that has not been well addressed. The major problems in measuring security metrics are two-fold: (1) *what to measure*, which is a question of how to define new, useful security metrics; and (2) *how to measure*, which asks how to devise new methods to measure security metrics. In this work, we are interested in answering the latter question, *how to measure* a security metric where the ground truth does not exist for the detection accuracy of a malware detector as well as for the class (i.e., malicious or benign) of the files.

When measuring the quality of malware detectors, many methods have been used based on certain heuristics such as using the labels of a few malware detectors as ground truth [1, 2, 3]. These heuristic-based approaches are troublesome because of the well-known fact that each malware detector has a different quality of detection accuracy. Although some methods have been proposed to answer *how to measure* security metrics [4, 5], measuring the relative accuracy of malware detectors has not been addressed in existing works. In particular, this work is inspired by our prior work [4] which measured the quality of malware detectors assuming that a voting-based estimation of detection accuracy is true. Unlike [4], this work aims to estimate the *relative* accuracy of malware detectors, which are obtained without making the assumptions that were made in [4]. Although relative accuracy of malware detectors are weaker than absolute accuracy of malware detectors, they are still useful in the process of decision making in regards to choosing malware detectors. Therefore, this paper aims at answering the following research question: "How can we rank the accuracy of malware detectors in the absence of ground truth?"

This work has the following **key contributions**: (i) This study offers a method to formulate how to estimate the *relative accuracy* of malware detectors. This method can be used when one needs to choose one malware detector over others; and (ii) The proposed algorithm measuring the relative detection accuracy of a malware detector is validated based on a real world malware dataset consisting of 62 detectors, given synthetic data with known ground truth values.

The rest of the paper is organized as follows. Section II provides the overview of the related state-of-the-art approaches. Section III presents a problem statement and our proposed methodology to solve the given problem. Section IV describes the experimental setup and results, with the discussion of key findings. Section V concludes the paper and suggests future research directions.

II. Related Work

In practice, obtaining ground truth is highly challenging and almost not feasible due to high noise, uncertain data, and/or the inherent nature of imperfect estimation. For example, when using machine learning techniques to train cyber defense models, we need to know the ground truth labels of training samples. For a small set of samples, we can use human experts to grade them and derive their ground truth. However, even in this case, the perfect evaluation is not guaranteed because humans are also error-prone and can make mistakes due to their inherent cognitive limitations. Accordingly, for a large set of samples, it is obviously not feasible for human experts to derive the ground truth. Therefore, it is true that many third-party datasets (e.g., blacklisted websites [6, 7, 8, 9]) are not necessarily trustworthy due to these inherent limitations.

Several studies have been conducted in order to investigate the accuracy of malware detectors when there exists no ground truth of their accuracy [10, 4, 5, 11]. These studies used different assumptions in order to estimate the accuracy of the malware detectors. **(author?)** [5] used the naïve Bayesian method and treated the unknown ground truth labels as hidden variables, while the Expectation-Maximization (EM)

This work is done when Jin-Hee Cho was with US Army Research Laboratory.

method [10, 11] is used to estimate the accuracy of malware detectors using well known metrics such as false-positive rate, false-negative rate, or accuracy. In [5], the authors assumed the homogeneity of false positives in all detectors, an independent decision of each detector, and low false positives but high false negatives assumed for all detectors, and so forth. However, these assumptions should be removed in order to reflect real world applications. **(author?)** [4] used a *frequentist* approach to design a statistical metric estimator to measure the quality metrics of malware detectors with only two of the four assumptions made in [5]. All the above works [10, 4, 5, 11] make certain assumptions. In contrast, the present paper investigates relative accuracy of malware detectors without making those assumptions.

The paper falls into the study of security metrics, which is an integral part of the Cybersecurity Dynamics framework [12, 13] and indeed one of the most fundamental open problems that have yet to be adequately tackled [14]. Recent advancement in security metrics includes [14, 15, 16, 17, 18, 19]. For example, the effectiveness of firewalls and DMZs is studied in [18] and the effectiveness of enforcing network-wide software diversity is investigated in [17]. Orthogonal to security metrics research are first-principle modeling of cybersecurity from a holistic perspective [20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30] and cybersecurity data analytics [31, 32, 33, 34, 35].

III. Problem Statement and Methodology

A. Definitions of Relative Accuracy of Malware Detectors

Suppose that there are m files, denoted by $F_1,\ldots,F_j,\ldots,F_m$ and n malware detectors, denoted by $D_1,\ldots,D_i,\ldots,D_n$. The input dataset is represented by a matrix $V=(V_{ij})_{1\le i\le n,1\le j\le m}$, which is defined where

$$V_{ij}=\begin{cases}1 & \text{if } D_i \text{ detects } F_j \text{ as malicious,}\\ 0 & \text{if } D_i \text{ detects } F_j \text{ as benign,}\\ -1 & \text{if } D_i \text{ did not scan } F_j.\end{cases}$$

Vector $V_i=(V_{i1},\ldots,V_{ij},\ldots,V_{im})$, where $1\le i\le n$, represents the outcome of using detector D_i to label m files.

With respect to a set of n detectors, the relative accuracy of detector i, denoted by T_i for $1\le i\le n$, is defined over interval $[0,1]$, where 0 means the minimum degree (i.e., a zero degree) of *relative accuracy* while 1 indicates the maximum degree of *relative accuracy*.

Definition 1: (Properties of relative accuracy) For a given set of files and a fixed set of n detectors, the relative accuracy of detector i, denoted by T_i for $1\le i\le n$, is defined based on the labels assigned by the n detectors (including detector i itself). For simplicity, we define T_i ranged in $[0,1]$.

We stress that the *relative accuracy* metric does not measure the true accuracy or trustworthiness of detectors. For example, consider three detectors D_1, D_2, and D_3, with respective true accuracy, 90%, 80%, or 70%. In this work, our goal is *not* to measure the accuracy of each detector. Instead, based on the labels of files assigned by these three detectors, we are more interested in knowing which detector is more accurate than others, leading to generating ranks of the examined detectors. Based on our proposed methodology, we obtain their respective relative accuracy as $T_1=100\%$, $T_2=90\%$, and $T_3=70\%$, which gives the performance of relative accuracy: detector $D_1>D_2>D_3$. However, the relative accuracy does not approximate the true accuracy. Moreover, for example, when we consider a set of files scanned by detectors, D_1, D_2, D_3, and D_4, letting the true accuracy of D_1-D_3 remain the same while the true accuracy of D_4 is 95%. Then, the resulting relative accuracy may be $T_1=80\%$, $T_2=60\%$, $T_3=50\%$, and $T_4=100\%$. This can be interpreted as the order of relative accuracy being $D_4>D_1>D_2>D_3$. This happens because the detector with the highest *relative accuracy* is always normalized to have a relative accuracy of 100% and the relative accuracy is always measured over a set of detectors.

B. Methodology

The basic underlying idea of estimating the relative accuracy of malware detectors is to measure the similarity between each pair of detectors. To do so, we iteratively create the relative accuracy of the malware detectors, given that the initial relative accuracy of each detector is set to 1, assuming that each detector is equally accurate.

1) Similarity Matrix: To measure the relative accuracy of detectors, the concept of a *similarity matrix* is introduced to collectively represent the similarity between malware detectors according to their decisions in labeling files as benign or malicious. In this matrix, denoted by $\mathbf{S}=(S_{ik})_{1\le i,k\le n}$, the i-th row corresponds to detector D_i and the k-th column corresponds to detector D_k, where $1\le i,k\le n$. Element S_{ik} denotes the similarity between detectors D_i and D_k in terms of their capabilities in detecting malware. Naturally, we require (i) $S_{ik}=S_{ki}$ because the similarity should be symmetric; and (ii) $S_{ii}=1$ for any $1\le i\le n$. Intuitively, the similarity between D_i and D_k, S_{ik}, is defined by the ratio of the number of decisions where D_i and D_k agree with each other over the total number of files scanned by both detectors, D_i and D_k.

To clearly define S_{ik} in a modular fashion, two auxiliary matrices are defined: the *agreement matrix*, denoted by $\mathbf{A}=(A_{ik})_{1\le i,k\le n}$, and the *count matrix*, denoted by $\mathbf{C}=(C_{ij})_{1\le i,k\le n}$. Intuitively, A_{ik} is the number of files upon which detectors D_i and D_k give the same labels, namely

$$A_{ik}=A_{ki}=\sum_{\ell=1}^{m}\begin{cases}1 & \text{if } V_{i\ell}=V_{k\ell}\wedge V_{i\ell}\neq -1\wedge V_{k\ell}\neq -1\\ 0 & \text{if } V_{i\ell}\neq V_{k\ell}\vee V_{i\ell}=-1\vee V_{k\ell}=-1.\end{cases}$$

and C_{ik} is the number of files scanned by both detectors, D_i and D_k, namely

$$C_{ik}=C_{ki}=\sum_{\ell=1}^{m}\begin{cases}1 & \text{if } V_{i\ell}\neq -1\wedge V_{k\ell}\neq -1,\\ 0 & \text{if } V_{i\ell}=-1\vee V_{k\ell}=-1..\end{cases}$$

Note that both $\mathbf{A}$ and $\mathbf{C}$ are symmetric. Given matrices $\mathbf{A}$ and $\mathbf{C}$, a similarity matrix $\mathbf{S}$ is defined as:

Definition 2: (Similarity matrix) The similarity matrix $\mathbf{S}=(S_{ik})_{1\le i,k\le n}$ is is defined as the ratio of labels that detectors D_i and D_k agree with each other, namely $S_{ik}=\frac{A_{ik}}{C_{ik}}$, implying that S_{ik} is symmetric.

2) Algorithm for Computing Relative Accuracy: Definition 1 specifies the properties that a good relative accuracy definition should meet. Now we address a specific definition to measure the relative accuracy that satisfies those desired properties; the definition is shown in Algorithm 1.

Algorithm 1 Computing relative accuracy

Input: Similarity matrix $\mathbf{S}_{n\times n}$; tolerable error threshold ε
Output: Relative accuracy vector $\mathbf{T} = [T_1, T_2, \ldots, T_n]^\mathsf{T}$

$\delta \leftarrow 2\varepsilon$
$\mathbf{T} \leftarrow ([1,1,\ldots,1]_{1\times n})^\mathsf{T}$
$\mathsf{NextT} \leftarrow ([0,0,\ldots,0]_{1\times n})^\mathsf{T}$
while $\delta > \varepsilon$ **do**
  $\mathsf{NextT} \leftarrow S \times T$
  $\mathsf{NextT} \leftarrow \mathsf{NextT}/\max(\mathsf{NextT})$
  $\delta \leftarrow \sum_{1\le i\le n} |T_i - \mathsf{NextT}_i|$
  $\mathbf{T} \leftarrow \mathsf{NextT}$
end while
Return T

The underlying idea of Algorithm 1 is as follows: The relative accuracy vector $\mathbf{T}$ is recursively calculated from the similarity matrix $\mathbf{S}$. The algorithm halts when the error δ is smaller than a threshold ε.

The similarity matrix $\mathbf{S}$ resembles a well-known correlation matrix consisting of correlation coefficients between a group of random variables. The major difference between these two kinds of matrices is that similarities are in the range of $[0,1]$ while correlations are in the range of $[-1,1]$. The sample version of a correlation matrix is the base for a statistical technique called *principal component analysis* where the *eigendecomposition* of the sample correlation matrix is used to find the dominating directions of variation in the data. In a similar sense, we use the similarity matrix to rank the relative accuracies of detectors. On the other hand, the recursive computation of the relative accuracy vector $\mathbf{T}$ may be reminiscent of a Markov Chain of n states. However, the similarity matrix $\mathbf{S}$ is not a probability transition matrix because the entries do not reflect probability.

IV. Experiments and Results

In this section, we conduct experiments using a synthetic dataset of known ground truth to evaluate the approach and then use the approach to analyze a real dataset.

A. Experiments with Synthetic Data

Generating synthetic data. Three synthetic datasets of labels are generated for one million samples per dataset: **D1** contains 300,000 malicious files and 700,000 benign files; **D2** contains 500,000 malicious files and 500,000 benign files; and **D3** contains 700,000 malicious files and 300,000 benign files. Using these three datasets allows us to see the impact of the ratio between malicious and benign files.

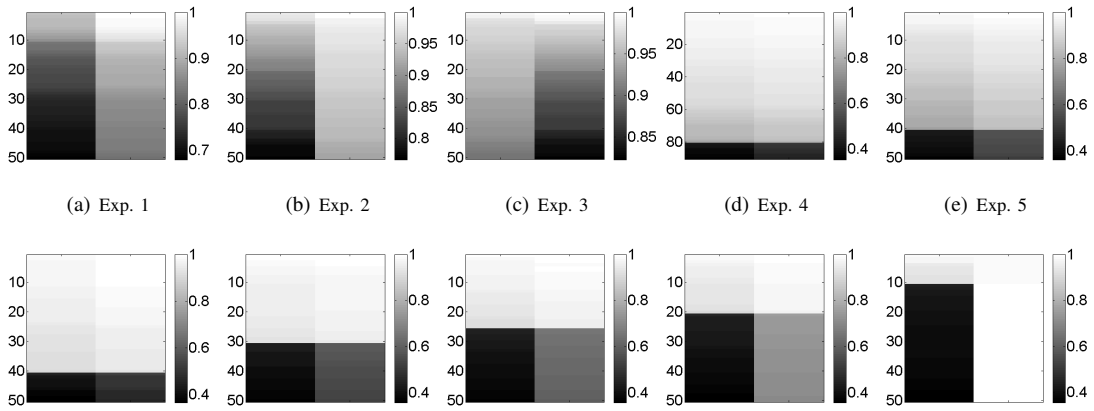
Experimental setup. We consider 10 experiments where each experiment uses a number of detectors characterized by a true-positive rate (TP) and a true-negative rate (TN), while false-positive rates (FP) and false-negative rates (FN) are used to derive TP and TN, such as $TP = 1 - FN$ and $TN = 1 - FP$ [14]. Moreover, accuracy is defined as $\frac{\text{TP+TN}}{\text{TP+FP+TN+FN}}$ [14].

Experiments 1-5 aim to test a variety of situations with various distributions of accuracies of malware detectors.

- **Experiment 1** - Four sets of detectors of varying true accuracy rates are simulated as:
  - 10 detectors with an accuracy range of 95% to 85%;
  - 10 detectors with an accuracy range of 85% to 75%;
  - 10 detectors with an accuracy range of 80% to 70%;
  - 20 detectors with an accuracy range of 75% to 65%.
- **Experiment 2** - Four sets of detectors of varying true accuracy rates are simulated as:
  - 10 detectors with an accuracy range of 100% to 90%;
  - 10 detectors with an accuracy range of 95% to 85%;
  - 10 detectors with an accuracy range of 90% to 80%;
  - 20 detectors with an accuracy range of 85% to 75%.
- **Experiment 3** - The algorithm is tested with all 50 detectors that have equal trustworthiness (i.e., a same detection capability) as:
  - 50 detectors with an accuracy range of 100% to 90%.
- **Experiment 4** - The algorithm is tested with *poor* detectors that have their accuracies below 50%:
  - 50 detectors with an accuracy range of 100% to 90%;
  - 10 detectors with an accuracy range of 95% to 85%;
  - 10 detectors with an accuracy range of 90% to 80%;
  - 10 detectors with an accuracy range of 85% to 75%.
  - 10 detectors with an accuracy range of 45% to 35%.
- **Experiment 5** - The algorithm is tested with a higher ratio of *poor* detectors as:
  - 10 detectors with an accuracy range of 100% to 90%;
  - 10 detectors with an accuracy range of 95% to 85%;
  - 10 detectors with an accuracy range of 90% to 80%;
  - 10 detectors with an accuracy range of 85% to 75%;
  - 10 detectors with an accuracy range of 45% to 35%.

Experiments 6-10 are conducted to investigate the threshold where the algorithm is able to diagnose 'good' detectors over 'poor' detectors as the ratio of the good detectors to the poor detectors decreases. In all 5 experiments, the good detectors have accuracies that range from 100% to 90%, while the poor detectors have accuracies that range from 45% to 35%.

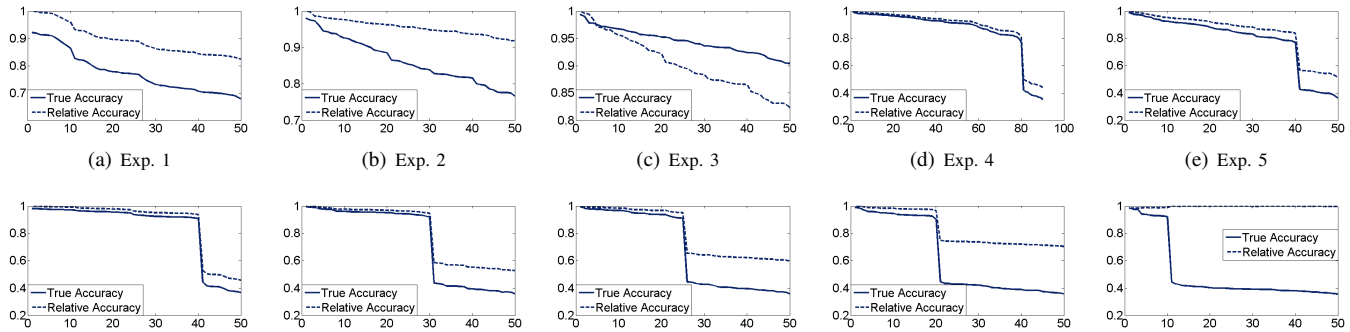
- **Experiment 6** - The algorithm is tested with 20% poor detectors as:
  - 40 detectors with an accuracy range of 100% to 90%;
  - 10 detectors with an accuracy range of 45% to 35%.
- **Experiment 7** - The algorithm is tested with 40% poor detectors as:
  - 30 detectors with an accuracy range of 100% to 90%;
  - 20 detectors with an accuracy range of 45% to 35%.
- **Experiment 8** - The algorithm is tested with 50% poor detectors as:
  - 25 detectors with an accuracy range of 100% to 90%;
  - 25 detectors with an accuracy range of 45% to 35%.



(a) Exp. 1 (b) Exp. 2 (c) Exp. 3 (d) Exp. 4 (e) Exp. 5

(f) Exp. 6 (g) Exp. 7 (h) Exp. 8 (i) Exp. 9 (j) Exp. 10

Fig. 1. Experiment results with dataset **D1**: In each picture, the *y*-axis corresponds to the detectors in an experiment while the *x*-axis corresponds to the True Accuracy (left-hand half) and Relative Accuracy (right-hand half) of the detectors in color scale.



(a) Exp. 1 (b) Exp. 2 (c) Exp. 3 (d) Exp. 4 (e) Exp. 5

(f) Exp. 6 (g) Exp. 7 (h) Exp. 8 (i) Exp. 9 (j) Exp. 10

Fig. 2. Experiment results with dataset **D1**: In each picture, the *y*-axis corresponds to the true accuracy and relative accuracy of each detector. The *x*-axis counts the individual detectors.

- **Experiment 9** - The algorithm is tested with 60% poor detectors as:
  - 20 detectors with an accuracy range of 100% to 90%;
  - 30 detectors with an accuracy range of 45% to 35%.
- **Experiment 10** - The algorithm is tested with 80% poor detectors as:
  - 10 detectors with an accuracy range of 100% to 90%;
  - 40 detectors with an accuracy range of 45% to 35%.

Fig. 1 plots experimental results with **D1**. For each experiment, we look into the *true accuracy* and *relative accuracy* of each detector. We observe that except Exp. 10, the order of the accuracies (e.g., detector 2 is more accurate than detector 3) is preserved by the relative accuracy (i.e., detector 2 has a higher relative accuracy than detector 3). This does not hold for Exp. 10 because 40 (out of the 50) detectors are 'poor.'

Fig. 2 maps both the *true accuracy* and the *relative accuracy* of each detector for the experiments in **D1**. Each experiment shows that there is one detector whose relative accuracy is 100%, which is implied by Algorithm 1. We also observe that the relative accuracy of a detector is *not* the same as the accuracy of the detector; in contrast, it can deviate significantly. However, note that the graph of the *relative accuracy* matches with the changes observed in the graph of the *true accuracy*, with minor differences in the slope and the degree of the change. This proves that recovering the true accuracy of each detector should be possible, if a proper method is provided. Experiment 10 is the exception again, showing that the rating system was overwhelmed by the 'poor' detectors which outranked the 'good' by the factor of 4 to 1. Furthermore, Experiments 6 through 9 show that with similar graphs for true accuracy, the difference in measurements between true accuracy and relative accuracy increases as the level of uncertainty increases.

For **D2** and **D3**, the results are almost identical to **D1**. The changes in the file distribution changed the initial ordering, but the results for comparing the true accuracy with the relative accuracy are trivial, so we didn't include them here.

Algorithm 1 provides whether the (true) accuracy of detectors is continuously distributed across a wide range (as in Experiments 1 and 2), distributed across a narrow range (as in Experiment 3), or distributed across a wide range in a discontinuous fashion (as in Experiments 4 and 5). Experiments 6 through 10 show that through all three datasets,

Algorithm 1 provides reliable results up to the point where there are 4 poor detectors per good detector. At this threshold, the poor detectors begin to be rated above the good detectors due to the noise introduced by sheer numbers.

Summarizing the experiment results with synthetic datasets **D1**-**D3**, we obtain the following insight:

Insight 1: Algorithm 1 is useful because it can compute the relative accuracy, or relative ranking, of malware detectors as long as the number of 'poor' detectors is not overwhelming.

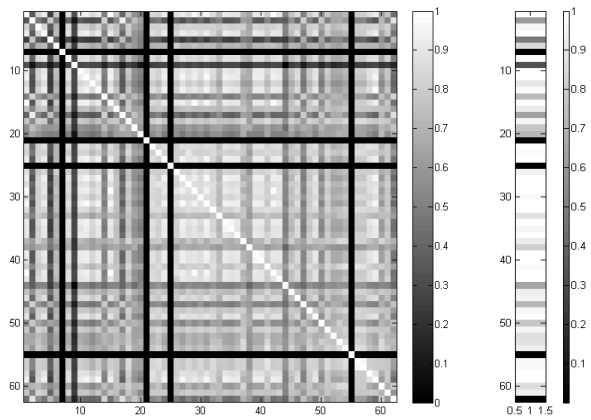
B. Applying the approach to evaluate a real dataset

The dataset was collected from VirusTotal. It contains a corpus of $m \approx 10.7$ million files, each of which was scanned by up to $n = 62$ anti-malware detectors, but some files were not scanned by every detector. Each detector labels a file it scanned as malicious ("1") or benign ("0"). The dataset is transformed to matrix $\mathbf{V_{ij}}_{n \times m}$, from which we derive a similarity matrix $\mathbf{S}$ and a relative accuracy vector $\mathbf{T}$ according to Algorithm 1.

TABLE I
THE AV NAME AND TRUST VALUE FROM THE PROCESSED DATA FROM VIRUSTOTAL.

AV Name	Trust	AV Name	Trust
BitDefender	100%	Symantec	95.78%
Ad-Aware	99.85%	CAT-QuickHeal	95.46%
McAfee	99.63%	Panda	95.22%
GData	99.62%	Emsisoft	95.22%
Kaspersky	99.48%	Zillya	93.20%
AhnLab-V3	99.45%	TotalDefense	92.80%
VIPRE	99.39%	nProtect	92.64%
MicroWorld-eScan	99.38%	Kingsoft	91.74%
Avast	99.37%	Bkav	91.68%
AVG	99.34%	Avira	89.93%
F-Pro	99.32%	Jiangmin	88.99%
K7AntiVirus	99.30%	TheHacker	88.33%
NANO-Antivirus	99.22%	Tencent	86.53%
F-Secure	99.05%	ViRobot	86.53%
McAfee-GW-Edition	99.04%	ALYac	85.91%
DrWeb	98.98%	Malwarebytes	79.80%
ESET-NOD32	98.78%	SUPERAntiSpyware	77.63%
Sophos	98.63%	ClamAV	77.47%
VBA32	98.45%	Baidu-International	73.13%
Comodo	98.45%	Qihoo-360	71.68%
Ikarus	98.44%	CMC	70.72%
AVware	98.31%	Zoner	70.17%
Fortinet	97.79%	Norman	65.44%
Cyren	97.53%	ByteHero	64.01%
TrendMicro	97.43%	AegisLab	60.10%
Microsoft	97.27%	Alibaba	47.14%
TrendMicro-HouseCall	97.03%	Arcabit	32.09%
Antiy-AVL	96.87%	AntiVir	5E-04%
Agnitum	96.73%	Commtouch	5E-04%
K7GW	96.50%	DrWebSE	3E-04%
Rising	95.81%	TotalDefense2	2E-06%

Table I summarizes the relative accuracy of the 62 detectors. We observe that the relative accuracy of 35 detectors is in $[1, 0.95]$, 11 detectors in $[0.85, 0.95]$, 7 detectors in $[0.7, 0.8]$, 3 detectors in $[0.6, 0.7]$, 1 detector in the 0.4 range, 1 detector in the 0.3 range, and 4 detectors at the order of magnitude of 10^{-6}. The extremely low relative accuracy of the last four detectors can be attributed to the following: (i) these detectors match poorly with the decisions of the other detectors; (ii) these detectors provide monotonous detection, meaning that they label all files either as 1 or 0; and (iii) these detectors scanned fewer than 1% of the files. Therefore, these detectors are correctly labeled as inaccurate.



(a) S (b) T

Fig. 3. Relative accuracy vector T and associated similarity matrix S.

Fig. 3 (a) shows the similarity matrix while Fig. 3 (b) shows the *relative* accuracy of the AV Detectors. The similarity matrix provides a good visual intuition as to why several detectors were rated low. The inherent symmetry is also observed. In order to reach the steady state, we ran 8 iterations of the algorithm to reach a resolution of $\varepsilon = 10^{-9}$.

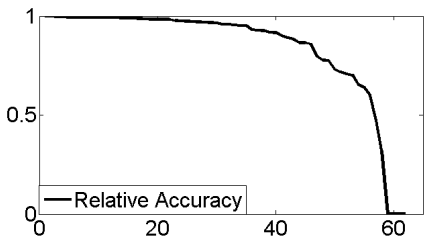


Fig. 4. Relative accuracy of the 62 detectors in the real-world dataset. Note that true accuracies of the detectors are not known and therefore not plotted.

Fig. 4 shows the relative accuracy of the 62 detectors in the real-world data. Summarizing the preceding discussion, we obtain the following insight:

Insight 2: A few detectors in the real-world dataset are not really useful. In traditional n-out-of-K voting, these poor detectors would be given equal weight votes with the good detectors. With the ability to discern which detectors are more reliable, more appropriate voting weights can be applied to the appropriate detectors.

V. CONCLUSION AND FUTURE WORK

We formulated the problem of estimating the *relative accuracy* of malware detectors in the absence of ground truth and presented an algorithm to derive the relative accuracy. We validated the proposed algorithm based on real-world datasets from VirusTotal, given synthetic data with ground truth. Through the extensive experimental study, we found that the proposed algorithm of estimating the relative accuracy of

malware detectors is capable of ranking the relative accuracies of the 62 real-world detectors which scanned millions of files. In particular, we identified 4 detectors that not only are useless, but also may do more harm than good.

We plan to conduct the following future research: (1) develop a theoretical evaluation framework that can be used to judge under what environments the proposed algorithm works or does not works; (2) characterize the co-variance and correlation between the accuracy of detectors; (3) develop an aggregation engine to incorporate detection labels of multiple malware detectors; and (4) identify key characteristics of poor detectors in order to avoid them when aggregating the labels of multiple detectors.

Acknowledgment. We thank VirusTotal for providing us the dataset. This work was supported in part by the U.S. National Science Foundation (NSF) under Grants CREST-1736209 and DMS-1620945, in part by the U.S. Department of Defense (DoD) through the Office of the Assistant Secretary of Defense for Research and Engineering, and in part by ARO under Grant W911NF-17-1-0566. The views and opinions of the authors do not reflect those of the NSF or DoD.

REFERENCES

[1] A. Mohaisen and O. Alrawi, "Av-meter: An evaluation of antivirus scans and labels," in *Proc. DIMVA*, pp. 112–131, 2014.

[2] J. Morales, S. Xu, and R. Sandhu, "Analyzing malware detection efficiency with multiple anti-malware programs," in *Proc. CyberSecurity'12*, 2012.

[3] R. Perdisci and M. U, "Vamo: Towards a fully automated malware clustering validity analysis," in *Proc. ACSAC'12*, pp. 329–338, 2012.

[4] P. Du, Z. Sun, H. Chen, J. H. Cho, and S. Xu, "Statistical estimation of malware detection metrics in the absence of ground truth," *IEEE Transactions on Information Forensics and Security*, pp. 1–1, 2018.

[5] A. Kantchelian, M. C. Tschantz, S. Afroz, B. Miller, V. Shankar, R. Bachwani, A. D. Joseph, and J. D. Tygar, "Better malware ground truth: Techniques for weighting anti-virus vendor labels," in *Proc. ACM AISec*, pp. 45–56, 2015.

[6] L. Invernizzi, S. Benvenuti, M. Cova, P. M. Comparetti, C. Kruegel, and G. Vigna, "Evilseed: A guided approach to finding malicious web pages," IEEE Symposium on Security and Privacy (2012), pp. 428–442.

[7] M. Kührer, C. Rossow, and T. Holz, "Paint it black: Evaluating the effectiveness of malware blacklists," in *Proc. RAID'14*, pp. 1–21.

[8] L. Xu, Z. Zhan, S. Xu, and K. Ye, "Cross-layer detection of malicious websites," in *Proc. ACM CODASPY'13*, pp. 141–152, 2013.

[9] J. Zhang, Z. Durumeric, M. Bailey, M. Liu, and M. Karir, "On the mismanagement and maliciousness of networks," in *Proc. NDSS'14*, 2014.

[10] A. P. Dawid and A. M. Skene, "Maximum likelihood estimation of observer error-rates using the em algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 20–28, 1979.

[11] V. C. Raykar, S. Yu, L. H. Zhao, A. Jerebko, C. Florin, G. H. Valadez, L. Bogoni, and L. Moy, "Supervised learning from multiple experts: Whom to trust when everyone lies a bit," in *Proc. ICML'09*, pp. 889–896, 2009.

[12] S. Xu, "Cybersecurity dynamics," in *Proc. HotSoS'14*, pp. 14:1–14:2, 2014.

[13] S. Xu, "Emergent behavior in cybersecurity," in *Proc. HotSoS'14*, pp. 13:1–13:2, 2014.

[14] M. Pendleton, R. Garcia-Lebron, J.-H. Cho, and S. Xu, "A survey on systems security metrics," *ACM Comput. Surv.*, vol. 49, pp. 62:1–62:35, Dec. 2016.

[15] S. Noel and S. Jajodia, *A Suite of Metrics for Network Attack Graph Analytics*, pp. 141–176. Cham: Springer International Publishing, 2017.

[16] J.-H. Cho, P. Hurley, and S. Xu, "Metrics and measurement of trustworthy systems," in *IEEE MILCOM 2016*, 2016.

[17] H. Chen, J. Cho, and S. Xu, "Quantifying the security effectiveness of network diversity: poster," in *Proc. HoTSoS'2018*, p. 24:1, 2018.

[18] H. Chen, J. Cho, and S. Xu, "Quantifying the security effectiveness of firewalls and dmzs," in *Proc. HoTSoS'2018*, pp. 9:1–9:11, 2018.

[19] E. Ficke, K. Schweitzer, R. Bateman, and S. Xu, "Characterizing the effectiveness of network-based intrusion detection systems," in *IEEE MILCOM*, 2018.

[20] X. Li, P. Parker, and S. Xu, "A stochastic model for quantitative security analysis of networked systems," *IEEE TDSC*, vol. 8, no. 1, pp. 28–43, 2011.

[21] S. Xu, W. Lu, and L. Xu, "Push- and pull-based epidemic spreading in arbitrary networks: Thresholds and deeper insights," *ACM TAAS*, vol. 7, no. 3, pp. 32:1–32:26, 2012.

[22] S. Xu, W. Lu, and Z. Zhan, "A stochastic model of multivirus dynamics," *IEEE TDSC*, vol. 9, no. 1, pp. 30–45, 2012.

[23] M. Xu and S. Xu, "An extended stochastic model for quantitative security analysis of networked systems," *Internet Mathematics*, vol. 8, no. 3, pp. 288–320, 2012.

[24] W. Lu, S. Xu, and X. Yi, "Optimizing active cyber defense dynamics," in *Proc. GameSec'13*, pp. 206–225, 2013.

[25] S. Xu, W. Lu, L. Xu, and Z. Zhan, "Adaptive epidemic dynamics in networks: Thresholds and control," *ACM TAAS*, vol. 8, no. 4, p. 19, 2014.

[26] Y. Han, W. Lu, and S. Xu, "Characterizing the power of moving target defense via cyber epidemic dynamics," in *Proc. HotSoS'14*, pp. 10:1–10:12, 2014.

[27] M. Xu, G. Da, and S. Xu, "Cyber epidemic models with dependences," *Internet Mathematics*, vol. 11, no. 1, pp. 62–92, 2015.

[28] S. Xu, W. Lu, and H. Li, "A stochastic model of active cyber defense dynamics," *Internet Mathematics*, vol. 11, no. 1, pp. 23–61, 2015.

[29] R. Zheng, W. Lu, and S. Xu, "Active cyber defense dynamics exhibiting rich phenomena," in *Proc. HotSoS'15*, pp. 2:1–2:12, 2015.

[30] R. Zheng, W. Lu, and S. Xu, "Preventive and reactive cyber defense dynamics is globally stable," *IEEE Transactions on Network Science and Engineering*, pp. 1–1, 2017.

[31] Z. Zhan, M. Xu, and S. Xu, "Characterizing honeypot-captured cyber attacks: Statistical framework and case study," *IEEE TIFS*, vol. 8, no. 11, pp. 1775–1789, 2013.

[32] Z. Zhan, M. Xu, and S. Xu, "Predicting cyber attack rates with extreme values," *IEEE TIFS*, vol. 10, no. 8, pp. 1666–1677, 2015.

[33] C. Peng, M. Xu, S. Xu, and T. Hu, "Modeling and predicting extreme cyber attack rates via marked point processes," *Journal of Applied Statistics*, vol. 0, no. 0, pp. 1–30, 2016.

[34] M. Xu, K. M. Schweitzer, R. M. Bateman, and S. Xu, "Modeling and predicting cyber hacking breaches," *IEEE TIFS*, vol. 13, no. 11, pp. 2856–2871, 2018.

[35] R. Garcia-Lebron, K. Schweitzer, R. Bateman, and S. Xu, "A framework for characterizing the evolution of cyber attacker-victim relation graphs," in *IEEE MILCOM*, 2018.